Contents lists available at ScienceDirect

Computational and Structural Biotechnology Journal

journal homepage: www.elsevier.com/locate/csbj

Mapping Cell Identity from scRNA-seq: A primer on computational methods

Daniele Traversa, Matteo Chiara *

Department of Biosciences, Università degli Studi di Milano, via Celoria 26, Milan 20133, Italy

ARTICLE INFO

Keywords:
ScRNAseq
Cell identity
Cell type annotation
Transcriptomics
RNAseq

ABSTRACT

Single cell (sc) technologies mark a conceptual and methodological breakthrough in our way to study cells, the base units of life. Thanks to these technological developments, large-scale initiatives are currently ongoing aimed at mapping of all the cell types in the human body, with the ambitious aim to gain a cell-level resolution of physiological development and disease. Since its broad applicability and ease of interpretation scRNA-seq is probably the most common sc-based application. This assay uses high throughput RNA sequencing to capture gene expression profiles at the sc-level. Subsequently, under the assumption that differences in transcriptional programs correspond to distinct cellular identities, *ad-hoc* computational methods are used to infer cell types from gene expression patterns. A wide array of computational methods were developed for this task. However, depending on the underlying algorithmic approach and associated computational requirements, each method might have a specific range of application, with implications that are not always clear to the end user. Here we will provide a concise overview on state-of-the-art computational methods for cell identity annotation in scRNA-seq, tailored for new users and non-computational scientists. To this end, we classify existing tools in five main categories, and discuss their key strengths, limitations and range of application.

1. Introduction

The term “cell” was first introduced by physicist Robert Hooke in 1665, in his work *Micrographia* [1], to describe the box-shaped structures he observed in cork using a microscope he self designed and perfected. This observation marks one of the earliest and most important concepts in the history of molecular biology.

In a way, the recent development of single cell (sc) technologies delineate another key technological breakthrough towards a better understanding of cells and their function. In the last decade sc-technologies, empowered the study of biological systems at the cellular level with a high throughput resolution and provided a refined characterization of cell types [2]. As the application of sc-based methods for the study of complex biological systems has become commonplace, large-scale initiatives such as the Human Cell Atlas and Tabula Sapiens, have been undertaken, with the ambitious goal to chart the human body at a single cell resolution [3,4].

Virtually any high throughput “omics” assay (genome, epigenome, transcriptome, proteome and others) can be combined with sc-technologies, both in isolation or simultaneously (see [5] for a comprehensive review) to obtain an integrated and comprehensive overview of the molecular landscape of a cell. This review will focus on scRNA-seq, which is currently considered the most mature and most commonly used sc-assay [6].

Since its introduction in 2009 [7] scRNA-seq has undergone constant improvements, through the development of novel technologies and library preparation strategies (see [8] for a comprehensive review). Irrespective of the methodological set-up, every scRNA-seq experiment is rooted in the same conceptual framework: the underlying assumption being that different cell types express different genes. Hence the primary objective of scRNA-seq is to identify/label the distinct cell types that populate a biological sample through the indirect observation of gene expression patterns.

In scRNA-seq library preparation molecular barcodes are used to flag NGS reads from a specific cell and transcript [9]. Dedicated tools such as cellRanger [10] and UMItools [11] are used in the analytical workflow to deconvolute barcodes and identify the genes expressed by each cell. Patterns of gene expression are collected in large counts tables, with cells on the columns and genes on the rows, that summarize the number of transcripts (reads) detected in every cell for every gene.

These tables are usually processed through highly flexible and powerful computational workflows such as Seurat [12] and Scanpy [13], which enable an efficient summarization of salient features and the identification of prospective cell types through the conceptual steps

* Corresponding author.
E-mail address: matteo.chiara@unimi.it (M. Chiara).

https://doi.org/10.1016/j.csbj.2025.03.051
Received 15 November 2024; Received in revised form 29 March 2025; Accepted 31 March 2025
Available online 2 April 2025

illustrated in the following section.

A common issue of scRNA-seq is that -due to inherent technological limitations- the number of genes/transcripts detected is limited, and usually only 1–5 % of the transcripts of a cell are observed [14]. It follows that counts tables in scRNA-seq are typically sparse, and the majority of the expression values (read counts) correspond to zero.

To mitigate the issue, cells with a limited number of detected expressed genes -usually below 200 [15,16]- are excluded from downstream analyses. Cell-cycle associated genes are often excluded as well, since in the context of incomplete or partial gene expression profiles their inclusion might skew cell type annotation towards different stages of the cell cycle [15,16].

Following data preprocessing and filtering, genes showing the largest changes in expression across the dataset are identified and used in subsequent analyses ("highly variable genes"). The underlying assumption being that the changes in expression observed for these genes should reflect the transcriptional programs of different cell types. Dimensionality reduction techniques, such as principal component analysis (PCA), are applied to the highly variable genes to summarise their pattern of expression in a discrete number of Principal Components (see [17] for a comprehensive overview of dimensionality reduction techniques used in scRNA-seq) and obtain a simplified representation of gene expression patterns. Unsupervised clustering is subsequently applied to gather cells with similar patterns of gene expression into clusters. We refer the reader to [18] for a comprehensive overview of clustering methods used in this domain of application.

Finally, each cluster is annotated with a cellular identity, by expert manual curation and/or by comparing genes associated with (or differentially expressed) in a cluster with lists of cell-type specific genes as available from the literature and/or through dedicated resources [2, 19–21].

The annotation of cell type identities represents a crucial step in scRNA-seq, and provides the ground for the biological interpretation of the results. While the workflow illustrated above represents *de facto* the standard in the field and was successfully applied for the discovery of novel cell types for several years [21] it also presents some important limitations.

For example clustering algorithms can form clusters only in the presence of a sufficient number of observations (the threshold is typically set by the user) and might overlook scarcely represented cell types. Additionally, since no clear/universally accepted rule or threshold is defined for the delineation of "highly variable genes", it is not always clear how many genes should be included in the analysis, and to what extent the patterns of expression of these genes might reflect different cell types, cell states and/or even random fluctuations in gene expression [20]. Another important limitation is that manual annotation of cell types is not completely reproducible and by definition can be biassed by the research question at hand and the domain knowledge of the investigator [19,20].

In an attempt to mitigate these limitations, many computational methods were developed to automate the annotation of cell types in scRNA-seq, without the need for manual curation, by building on collections of publicly available data with high quality annotations. These methods can annotate clusters of similar cells defined according to the protocol outlined above, but also each distinct cell in the dataset, one by one, without any need of clustering and selection of "highly variable genes". Although all these computational methods were developed for the same task they employ different computational strategies, use different algorithms and have different computational requirements, which might not always be obvious to the user.

This short review will provide a comprehensive yet concise overview of state of the art computational methods for the annotation of cellular identities in scRNA-seq, with the aim to illustrate clearly and in simple terms the key conceptual differences and their potential strengths and limitations to prospective new users and or non experts in the field.

1.1. A broad categorization of tools for cell type annotation in scRNA-seq

The key assumption of scRNA-seq is that different cell types express different genes. Building on this idea, methods for the automatic annotation of cell type identity capitalize on previous/publicly available knowledge to derive an accurate representation of characteristic patterns of gene expression and standardise cell type annotation. Hence, irrespective of the computational approach/algorithm used, knowledge bases with high quality annotations of cell types represent a crucial prerequisite.

Notwithstanding these common assumptions, contemporary methods for cell identity annotation in scRNA-seq are laid on different computational and conceptual frameworks.

At one end of the spectrum we find methods that rely on a fully deterministic approach and build on the concept of "marker gene". Marker genes are genes that are expressed primarily in a specific cell type [22]. In this framework cell types are "simply" defined by different (linear) combinations of marker genes.

At the other end of the spectrum, more elaborate models based on Machine Learning (ML) algorithms are trained on expert annotated datasets to consistently reproduce manual annotation on new data. Unlike marker genes based tools, ML-models are not constrained to a pre-defined set of genes. Instead, they extract patterns from gene expression data using a broader set of descriptive features. This flexibility allows ML-models to capture nonlinear patterns in gene expression which could be easily overlooked by more simple deterministic approaches. However, since the number and type of features used by different ML methods can vary, the essence of the decision process can not be easily summarized in a simple human-comprehensible equation or rule. As a result ML models are associated with a general loss in the transparency and interpretability [23].

ML is a wide and quickly developing field. Different ML algorithms build on different assumptions and core concepts and might even have different computational and hardware requirements. For example, recent breakthroughs in Deep Learning (see here [24]) represent a turning point in the field of ML, but the underlying technology and methodological implications are not always clear to non-expert in the field (detailed below).

Moreover ML-models can be trained using different strategies such as: supervised learning -where labelled data are presented to the model from the start and only a predetermined number of labels can be learned and semi-supervised learning -where only a subset of the example data is associated with a known label and information from labelled data is borrowed to classify the other data. Furthermore, in an attempt to obtain more powerful and accurate ML-models, different approaches and algorithms can be (and often are) integrated in a single computational method/classifier, adding further layers of complexity.

Considering the wealth of available algorithms and training strategies, the ideal range of application of ML methods might not always be obvious.

Guided by these considerations, here we compiled an up to date (as of August 2024) catalogue of computational methods for the automatic annotation of cell identities for scRNA-seq data by integrating the scRNA-seq-tools database [25], with comprehensive manual review of scientific literature. Methods were subsequently classified in 5 broad categories (Fig. 1) depending on the main computational approach: 1) marker-based (MB); 2) classical ML (C-ML); 3) semi-supervised learning (SSL); 4) deep learning (DL) and 5) Hybrid methods (HM).

In the following sections, to facilitate the comparison and aid prospective users in the selection of the most appropriate method for their specific task, we will offer an overview of the main strengths, limitations and ideal scope of application of each of these 5 categories. Further, the most important properties of every tool, along with its categorization will be summarized in 5 tables (Table 1 to Table 5).

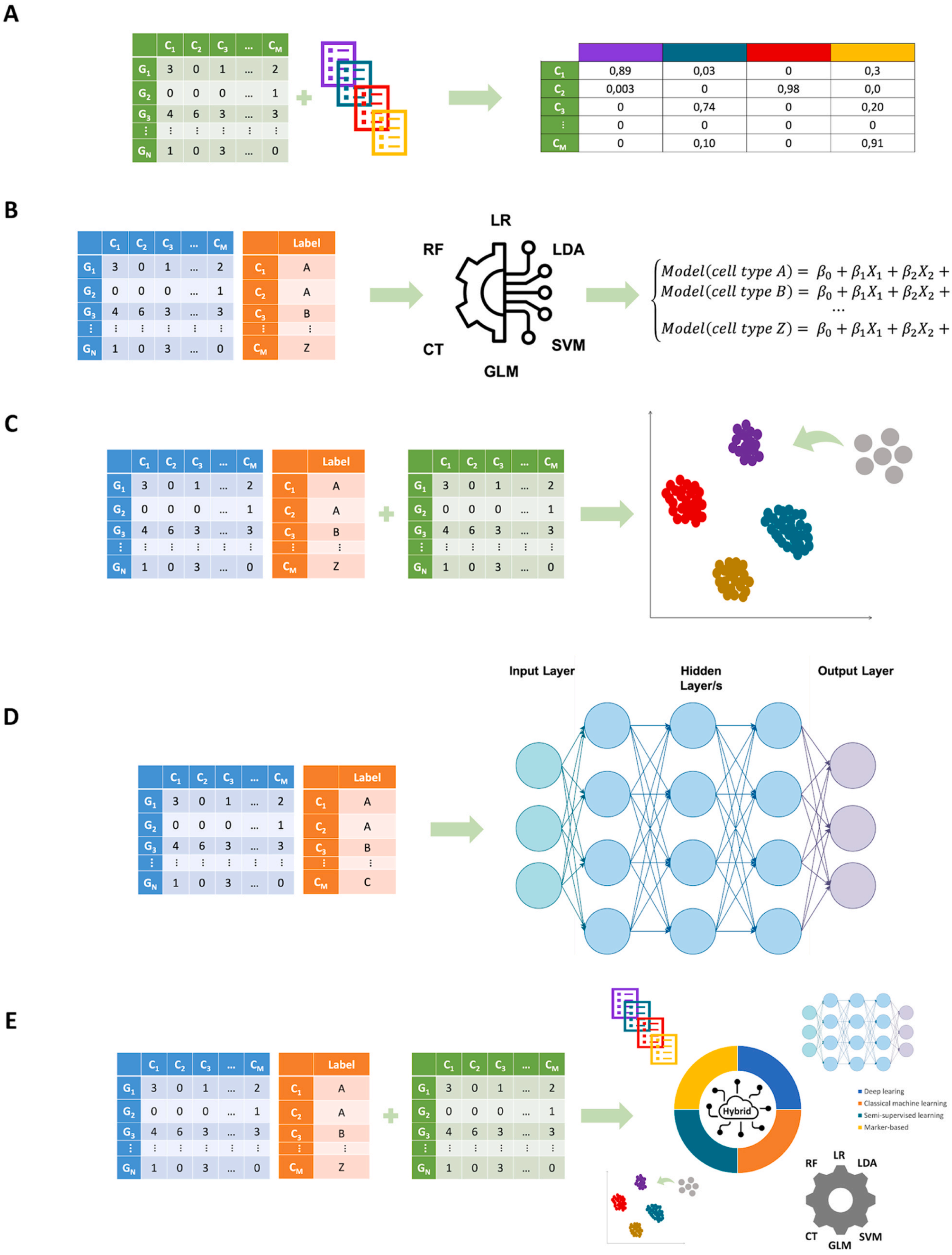

Fig. 1. Schematic representation of the five distinct main conceptual frameworks used for the implementation of computational methods for the classification of cell type identities from scRNA-seq data: **A)** marker-based (MB). Counts matrix of unlabelled cells (green table) are annotated through *ad hoc* scoring systems employing cell type specific lists of genes (indicated by different colours); **B)** classical machine learning (C-ML). Machine learning algorithms are trained with labelled (orange) counts matrix data (blue); **C)** semi-supervised learning (SSL). Labelled data (blue= count matrix and orange= annotation labels) and unlabeled data (green) processed in the same analytical workflow to transfer labels; **D)** deep learning (DL); a recent breakthrough in ML applies neural networks to learn labelled data (counts=blue, labels=orange) **E)** Hybrid methods any of A to D is combined in a single workflow.

Table 1
marker-based (MB) tools.

Tool	Impl	AL	#cited	Repository	Custom markers
ACTIONet [26]	Python/R	cell	53	http://github.com/decomverse/ACTIONet	Yes
adobo [27]	Python	cluster	37	http://oscar-franzen.github.io/adobo/	Yes
Besca [28]	Python	cluster	19	http://bedapub.github.io/besca/	Yes
CIA [29]	Python	cell	0	http://github.com/ingmbioinfo/cia	Yes
CellAssign [30]	R	cell	320	http://github.com/irrationone/cellassign	Yes
CellMeSH [31]	Python	cluster	11	http://github.com/shunfumao/cellmesh	No
deCS [32]	R	cluster	19	http://github.com/bsm1320/deCS	No
DigitalCellSorter [33]	Python	cluster	30	http://github.com/sdomanskyi/DigitalCellSorter	Yes
MarkerCount [34]	Python	both	14	http://github.com/combio-dku/MarkerCount/tree/master	Yes
MACA [35]	Python	both	19	http://github.com/ImXman/MACA	No
Sargent [36]	R	cell	11	NA	No
scCATCH [37]	R	cluster	215	http://github.com/ZJUFanLab/scCATCH	No
SCINA [38]	R	cell	184	http://github.com/jcao89757/SCINA	Yes
scMayoMap [39]	R	cluster	10	http://github.com/chloelulu/scMayoMap	Yes
scMiko [40]	R	cluster	12	http://github.com/NMikolajewicz/scMiko	Yes
scMAGIC [41]	R	cell	15	http://github.com/TianLab-Bioinfo/scMAGIC	No
scMRMA [42]	R	cluster	28	http://github.com/JiaLiVUMC/scMRMA	Yes
scQCEA [43]	R	cell	3	http://isarnassiri.github.io/scQCEA/	No
SCSA [44]	Python	cluster	122	http://github.com/bioinfo-ibms-pumc/SCSA	Yes
scSorter [45]	R	cell	91	http://github.com/cran/scSorter	Yes
scType [46]	R	cluster	413	http://github.com/IanevskiAleksandr/sc-type	Yes
scTyper [47]	R	cluster	24	http://github.com/omicsCore/scTyper	No
Garnett [48]	R	cell	447	http://cole-trapnell-lab.github.io/garnett/	No

Tool: tool name; **Impl**:programming language/languages used for the implementation; **AL:** annotation level the tool annotates at single cell resolution; cluster: the tool can annotate only predefined clusters of cells; both: both single cell and cluster; **#cited:** number of times cited; **repository:** hyperlink to the software repository; **Custom markers:** yes if the tool allows the specification of custom markers, no otherwise

1.2. Marker-based (MB)

Marker-based (MB) methods (Table 1 [26–48]) implement mathematical models or custom scoring systems to annotate cell types based on a predefined selection of cell-type specific marker genes (Fig. 1 A).

The definition of an optimal list of marker genes is probably the most crucial requirement for the development of marker-based methods [22]. Several approaches could be used to this end, including lists of marker genes collected from the literature; dedicated publicly available repositories such as CellMarker [49]; and/or construction of custom lists of markers through the analysis of expert curated annotations of cell types (as those available from the Human Cell Atlas [3] or PangLaoDB [50]). Importantly, most MB-methods also allow users to define custom cell types and sets of marker genes, which improves the potential for the annotation of novel/additional cell types.

Indeed, the fact that any novel list of marker genes could in theory be evaluated and scored, without the need of training and/or calibration [22], probably represents the main advantage of MB-tools over ML and SSL methods. Since in general MB-based tools are agnostic w.r.t the biological function/meaning of markers it is also theoretically possible to discretise/refine annotations by providing lists of genes that delineate

Table 2
classical machine learning (C-ML) based tools.

Tool	Impl	AL	#cited	Repository
CaSTLe [51]	R	cell	127	http://github.com/yuvallb/CaSTLe
CHETAH [52]	R	cell	263	http://github.com/jdekanter/CHETAH
CellO [53]	Python	cluster	33	http://github.com/deweylab/CellO
ELeFHAnt [54]	R	both	0	http://github.com/praneet1988/ELeFHAnt
HieRFIT [55]	R	cell	7	http://github.com/yasinkaymaz/HieRFIT
IBMR [56]	R	cell	4	http://github.com/keshav-motwani/IBMR
scAnnotatR [57]	R	cell	19	http://github.com/grisslab/scAnnotatR
scDetect [58]	R	cell	4	http://github.com/IVDgenomicslab/scDetect
SciBet [59]	R/C+ +	cell	142	http://github.com/PaulingLiu/scibet
scPred [60]	R	cell	376	http://github.com/powellgenomicslab/scPred
SingleCellNet [61]	R	cell	335	http://github.com/CahanLab/singleCellNet
Moana [62]	Python	both	79	http://github.com/yanailab/moana
scASK [63]	Matlab	cell	2	http://github.com/liubo2358/scASKcmd

Tool: tool name; **Impl**:programming language/languages used for the implementation; **AL**: annotation level the tool annotates at single cell resolution; cluster: the tool can annotate only predefined clusters of cells; both: both single cell and cluster; **#cited:** number of times cited; **repository:** hyperlink to the software repository

Table 3
semi-supervised learning (SSL) based tools.

Tool	Impl	AL	#cited	Repository
CIPR [73]	R	cluster	70	http://github.com/atakanekiz/CIPR-Package
Capybara [74]	R	cell	48	http://github.com/morris-lab/Capybara
clustifyr [75]	R	both	107	http://ttps://github.com/rnabioco/clustifyR
northstar [76]	C+ +/ Python	cluster	15	http://github.com/fabilab/northstar
ProjectSVR [77]	R	cell	0	http://github.com/jarninggau/ProjectSVR
RCA2 [78]	R	cell	16	http://github.com/prabhakarlab/RCAv2
scAdapt [79]	Python	cluster	24	http://github.com/biomed-ai/scAdapt
scID [80]	R	cluster	56	http://github.com/BatadaLab/scID
scLearn [81]	R	cell	27	http://github.com/bm2-lab/scLearn
scmap [82]	R	Both	628	http://github.com/hemberg-lab/scmap-shiny
scMatch [83]	Python	cell	104	http://github.com/forrest-lab/scMatch
SingleR [84]	R	cell	3548	http://github.com/dviraran/SingleR
Sincast [85]	R	cell	5	http://github.com/meiosis97/Sincast
SPEEDI [86]	R	cell	3	http://github.com/FunctionLab/SPEEDI/
Symphony [87]	R/C+ +	both	176	http://github.com/immunogenomics/symphony

Tool: tool name; **Impl**:programming language/languages used for the implementation; **AL**: annotation level the tool annotates at single cell resolution; cluster: the tool can annotate only predefined clusters of cells; both: both single cell and cluster; **#cited:** number of times cited; **repository:** hyperlink to the software repository

different cell subtypes, states or properties. In these respects, MB-based methods are more flexible compared to ML-based methods and their range of applications can be more easily adapted to the user's specific needs.

Another important consideration is that the algorithms implemented by MB tools are completely deterministic and hence the results can be more easily assessed and inspected, compared to ML-based methods. Despite these attractive benefits, marker-based annotation also presents drawbacks which sometimes hamper their application. The most important limitation is that well characterised/validated marker genes are currently available only for a reduced number of cell types, and cell types lacking a characterization can not be annotated [21]. Additionally, since scRNA count tables are usually sparse, due to the low per cell sequencing depth, markers might not be systematically detected across every cell in the dataset [22]. Due to these considerations, annotation of cell types based on lists of marker genes might represent a double-edged sword since the accuracy strictly depends on the availability of reliable lists of marker genes and previous knowledge [21,22]. Additionally, lists of marker genes that are either too small or do not include a sufficient number of unique markers, might limit the resolution causing an increase of the proportion of unclassified or inaccurately classified cells.

1.3. 2- Classical ML (C-ML)

By "Classical machine learning (C-ML)" here we address the wide collection of computational ML methods which predate the recent technological breakthroughs of Deep Learning (DL) [23–25]. An incomplete list includes: support vector machines (SVMs), random forest (RF), logistic regression (LR) XGBoost among the others (we refer the reader to [23] for a more comprehensive overview). C-ML algorithms provide the computation core of methods for the annotation of cellular

Table 4
deep learning (DL) based tools.

Name	Impl	AL	#cited	Repository
ACTINN [88]	Python	cell	235	http://github.com/mafeiyang/ACTINN
CAMLU [89]	R	cluster	11	http://github.com/ziyili20/CAMLU
CellTICS [90]	Python	cell	7	http://github.com/qyyin0516/CellTICS
Cell BLAST [91]	Python	cell	121	http://github.com/gao-lab/Cell_BLAST
CIForm [92]	Python	cell	36	http://github.com/zhanglab-wbgcas/CIForm
JIND [93]	Python	cell	7	http://github.com/mohit1997/JIND
mtANN [94]	Python/ R	cell	3	http://github.com/Zhangxf-ccnu/mtANN
mtSC [95]	Python	cell	16	http://github.com/bm2-lab/mtSC
N-ACT [96]	Python	cell	1	http://ttps://github.com/SindiLab/N-ACT?tab=readme-ov-file
NeuCA [97]	R	cell	14	http://github.com/haoharryfeng/NeuCA?tab=readme-ov-file
CellLM [98]	Python	cell	17	http://github.com/PharMolix/OpenBioMed
OnClass [99]	Python	cell	50	http://github.com/wangshenguiuc/OnClass
Pollock [100]	Python	cell	2	http://github.com/ding-lab/pollock
scANNA [101]	Python	cell	1	http://github.com/SindiLab/scANNA
scBERT [102]	Python	cell	288	http://github.com/TencentAILabHealthcare/scBERT
scCapsNet [103]	Python	cell	52	http://github.com/wanglf19/scCaps
scDeepHash [104]	Python	cell	0	http://github.com/bowang-lab/scHash
scDeepInsight [105]	Python	cell	26	http://github.com/shangruJia/scDeepInsight
scDeepSort [106]	Python/ R	cell	105	http://github.com/ZJUFanLab/scDeepSort
scDHA [107]	R/C+ +	cell	127	http://github.com/duct317/scDHA
scGraph[108]	Python	cell	22	http://github.com/QijinYin/scGraph
scMRA [109]	Python	cell	29	http://github.com/ddb-qiwang/scMRA-torch
scPretain [110]	Python	cell	11	http://github.com/ruiyi-zhang/scPretrain
scRCA [111]	Python	cell	0	http://github.com/LMC0705/scRCA
scSHARP [112]	Python/ R	cell	4	http://github.com/lewinsohndp/scSHARP
Selina [113]	Python/ R	cell	4	http://github.com/wanglabtongji/SELINA.py
sigGCN [114]	Python	cell	53	http://github.com/NabaviLab/sigGCN
SIMS [115]	Python	cluster	4	http://github.com/braingeneers/SIMS
SuperCT [116]	Python/ R	cell	66	http://sct.lifegen.com/ (web app)
TOSICA [117]	Python	cell	126	http://github.com/JackieHanLab/TOSICA
scGPT [118]	Python	cell	415	http://github.com/bowang-lab/scGPT
MARS [119]	Python	cell	132	http://github.com/snap-stanford/mars

Tool: tool name; **Impl**:programming language/languages used for the implementation; **AL**: annotation level the tool annotates at single cell resolution; cluster: the tool can annotate only predefined clusters of cells; both: both single cell and cluster; **#cited:** number of times cited; **repository:** hyperlink to the software repository

Table 5
hybrid tools.

Tool	Impl	AL	#cited	Repository
GraphCS [124]	Python/ R/C+ +	cell	26	http://github.com/biomed-AI/GraphCS
LAmbDA [125]	Python	cell	48	http://github.com/tsteelejohnson91/LAmbDA
scIAE [126]	Python/R	cell	21	http://github.com/JGuan-lab/scIAE
scNym [127]	Python/R	cell	96	http://github.com/calico/scnym
TripletCell [128]	Python	cell	9	http://github.com/liuyan3056/TripletCell
scSimClassify [129]	Python	cell	5	http://github.com/digi2002/scSimClassify
UNIFAN [130]	Python	cluster	1	http://github.com/doraadong/UNIFAN
scAnCluster [131]	Python	cluster	45	http://github.com/xuebaliang/scAnCluster
CellMarkerAccordion [132]	R	both	3	http://github.com/TebaldiLab/cellmarkeraccordion
scAnnoX [133]	R	cell	4	http://github.com/XQ-hub/scAnnoX
CellGrid [134]	Python	cell	10	http://github.com/Brodinlab/cellgrid
gCAnno [135]	Python	cell	2	http://github.com/xjtu-omics/gCAnno
PopV [136]	Python	cell	7	http://github.com/YosefLab/popV
scClassify [137]	R	cell	119	http://github.com/SydneyBioX/scClassify
scHPL [138]	Python	cell	36	http://github.com/lcmmichielsen/scHPL
Azimuth[139]	R	cluster	10528	http://github.com/satijalab/seurat

Tool: tool name; **Impl**:programming language/languages used for the implementation; **AL**: annotation level the tool annotates at single cell resolution; cluster: the tool can annotate only predefined clusters of cells; both: both single cell and cluster; **#cited:** number of times cited; **repository:** hyperlink to the software repository

identity from scRNA-seq data (Table 2 [51–63]).

In C-ML, a machine learning classifier is trained using pre-labelled data (training set), to learn expert curated or high quality cell type annotations. Data are described through features, that is individual measurable properties computed from the data itself for every cell (for example: n° of expressed genes; level of expression of a specific gene; n° of expressed genes in a specific pathway or ontology).

The objective is to develop a computational model capable of consistently and automatically reproducing the annotation of the training set on novel unlabeled data based on the observation of the features (Fig. 1B). Following training, the performances of the C-ML methods are typically assessed on an independent test set, that is a dataset with known annotations/labels, but formed exclusively by new data which were not presented to the model during training. This strategy is used to measure the model's generalization and predictive accuracy (see section Benchmarking cell type annotation).

Training is a fundamental step in any machine learning application, and is common to both DL and C-ML. However, whereas in C-ML data are described through a discrete number of predetermined features selected and engineered by experts in the field [23], in DL millions of features are extracted and computed directly from the data.

Hence, since the need for the optimization of millions of features and parameters, training DL algorithms require massive datasets, is very computationally demanding and needs dedicated hardware such as GPUs or TPUs; conversely C-ML algorithms can be trained on limited numbers of training examples on virtually any hardware architecture [24,25]. Due to these reasons C-ML methods are often used for ablation studies [23], that is experiments where one or multiple features are removed from the input during training, and their impact is estimated a *posteriori* by recording the accuracy of the resulting model [23]. Another important property of C-ML methods is that the underlying decision model is more easily interpretable compared to the complex architectures implemented by DL (see below). For instance, simple linear relationships can be represented using linear models. A key limitation of C-ML models is that they perform optimally only when trained to recognize a limited number of labels/classes. Applied to scRNA-seq, this limits the type of discernible cell identities and their correct annotation. More importantly, only cell types that were originally included in the training set can be recognized and annotated, and the inclusion of new cell types requires a novel round(s) of training [23]. Another potential issue is that most C-ML methods require a balanced training set with similar numbers of data points for every class [23], a requirement that is not usually met by scRNA-seq datasets where extreme fluctuations in the abundance of each cell type can be observed.

The integration of multiple datasets could ideally represent a possible solution to this limitation. Unfortunately, datasets with consistent and homogenized cell type annotations for a large number of cells and across different tissues are rather an exception than a rule in the field of scRNA-seq. Moreover, similar to any large scale omics assay, also scRNA-seq is prone to batch effects caused by difference experimental settings and/or technical limitations, (reviewed in [64]) and the integration of data obtained under different conditions might result extremely challenging.

These considerations impact the generalization of ML models and represent a crucial limitation to the development of any method for the reproducible annotation of cell identity in scRNA-seq.

1.4. 3- Unsupervised and semi-supervised learning (SSL)

Unsupervised and semi-supervised learning (SSL) combines labelled and unlabelled data in a common analytical workflow with the aim to transfer annotations from a reference dataset (labelled data) to the new instances of the data (unlabelled data) through one or more a similarity metrics (Fig. 1C) [24].

Conceptually speaking SSL stands in a middle ground between MB and C-ML: labelled data are used to infer distinctive transcriptional patterns associated with known cell identities; then similarity metrics are used to impute annotations on the unlabeled data based shared patterns of gene expression.

Several SSL strategies and algorithms have been adapted to cell type classification in the recent past [65–69], including for example: supervised clustering [70], similarity graphs [68], and pairwise comparison to reference annotations. In the same way a variety of different similarity metrics were applied to reconcile gene expression patterns, including (but not limited to) correlation, change in expression (logFC) of differentially expressed genes and many others [70–72]. Table 3 reports the complete list of SSL based methods for cell type annoation [73–87].

SSL tools provide several attractive features compared to MB and C-MB since: 1) the underlying results are easily interpretable (similar to MB); 2) no-predefined set of marker genes is required (similar to C-ML) and the complete transcriptome is evaluated at glance. However, SSL methods are affected by virtually all the limitations discussed previously for C-ML methods. Also in this case the selection of a reference dataset/reference datasets for the construction of the model is crucial and ultimately determines its range of application. Indeed, by definition only cell types for which an adequate number of labelled examples are available can be discerned and incomplete, error-prone and non accurately annotated reference data might result in error propagation.

Additionally, similarly to C-ML also in this case the inclusion of novel cell types, requires a continuous refinement/repurposing of the model and might become impractical to/for non experienced users [23]. Another point of similarity with C-ML, SSL may struggle to generalize and classify cell type for which a limited number of examples are provided in the reference dataset and/or in the presence of skewed or imbalanced data. Also in this case the availability of large, homogenized and consistently annotated collection of cell types from scRNA-seq would represent the ideal solution for the development of more general and powerful methods. However, as discussed above, these requirements might be out of reach for the next few years due to inherent technical limitations (see batch effect in the previous section).

1.5. 4-Deep-learning (DL)

Deep learning (DL) methods have recently emerged as powerful tools for the analysis of scRNA-seq since the ability to model complex, high-dimensional data and capture intricate, non-linear relationships in gene expression patterns (Fig. 1D). Table 4 provides a complete list of DL methods for cell identity annotation [88–119].

The rapid advancement of DL has been driven by a combination of technological and algorithmic developments that have significantly improved the performance, scalability, and accessibility of more elaborate and powerful ML architectures such as deep neural networks [24].

Neural networks (NN) are a class of machine learning models inspired by the structure and function of the human brain. NNs consist of layers of interconnected nodes (neurons) that process and transform data. Each neuron receives input data, which are weighted and summed to emit an output. Multiple layers of neurons can be arranged and interconnected to deconstruct data and learn patterns and the output of every neuron can also be weighted, depending on its importance in the decision process. The output layer -the final layer of a NN- collects the output from the other layers (inner-layers) and emits the final prediction. In simple terms, the process of training a neural network involves iteratively adjusting weights (of neurons and data) to optimize accuracy of the prediction of the model.

Deep neural networks (DNNs) are neural networks with more complex architectures, typically consisting of multiple hidden layers (whereas standard neural networks often have just one or two [23]). Due the increased complexity in their architecture DL models require larger dataset for training, and are significantly more computationally intensive compared to C-ML models. Consequently, dedicated hardware such as for example TSU and/or GPUs is needed. While these computational requirements represent a limitation for the construction of DL models, it also enables DL to process extremely large datasets, that would be computationally unfeasible for C-ML. Additionally, since DL performs optimally with large amounts of data, their accuracy is bound to increase as more data become available [120].

Differences in the implementation aside, the process of developing DL models is conceptually analogous to that of C-ML algorithms: the model learns from accurately labeled training data. Thus selecting a good reference dataset for training remains a critical step. Hence, the limited availability of consistently annotated reference datasets and challenges in the integrations of different datasets due to batch effects, pose a challenge also for DL.

However, since DL leverages sophisticated feature decomposition techniques such as autoencoders and transformers (see [121] for a more comprehensive discussion), which can effectively reduce noise and capture meaningful patterns in the data, in the presence of an adequately large body of data, DL based methods are more resilient to batch effect and noise compared to DL.

Since their intricate architecture and numerous layers of abstraction DL models are often regarded black boxes with results that are hard to explain. To mitigate these limitations DL methods can be specifically deployed for feature importance or feature attribution analysis (reviewed in [122]). These techniques help unravel the contribution of individual features or neurons in the model's decision-making process, offering valuable insights into the data. Such methods are particularly useful for reconciling the results with biological knowledge and have been used in genomics, healthcare, and, more recently, in single-cell RNA sequencing (scRNA-seq) analysis [123].

1.6. 5- Hybrid methods (HM)

Hybrid methods (HM, Table 5 [124–138]) integrate different algorithms or computational approaches to leverage their complementary strengths and build methods with enhanced performance (Fig. 1E). For example, hybrid methods for cell type annotation in scRNA-seq might combine and integrate different SSL and C-ML approaches and/or even integrate data across different training sets to improve accuracy and generalizability. Since each distinct method builds on different assumptions and integrates different algorithms, HM represents a very broad category. Nevertheless all the methods in this category share some principles and commonalities.

Irrespective of the implementation, a key requirement for any HM is the need to efficiently combine and integrate results from different methods into a single prediction. This can be achieved through various approaches, ranging from simple majority consensus (where the final prediction is assigned based on the most frequently predicted label) to more sophisticated approaches that train an ad-hoc ML-model for label integration. Consensus based strategies help reduce bias, mitigate errors, and improve the overall reliability of cell type classification, particularly in the context of complex, high-dimensional data [23]). However, these improvements come at the cost of increased computational demands, since the need to train and optimize multiple models. Moreover- another crucial consideration- the implementation of the consensus strategy is not banal and might require continuous refinement and manual curation as new classes/data labels are introduced. Similar to any other ML based methods, also HM can correctly detect/classify only the labels that were observed in the training set, and the addition of new classes requires further rounds of training. In HM this problem is exacerbated by the need to train multiple models. As a consequence the choice of the reference dataset used for training and development remains the most critical point. In summary, although consensus based approaches might enhance the predictive power of a tool, they are still subject to the same limitations that affect the underlying methods they integrate. Finally, since multiple independent methods contribute to the final prediction, the explainability of HM tools is generally lower compared to methods based on a single predictor/approach.

1.7. Benchmarking cell type annotation

Several recent studies engaged in the task of comparing the effectiveness of a selection of the computational methods illustrated in this review (and by extension of the underlying computational approach) by evaluating their performances in a controlled environment with carefully selected high quality annotated data.

Since cell identity annotation is a classification problem, where the goal is to attach the correct label to the correct cell, metrics commonly used for the assessment of ML-classifiers are employed also for the comparison of methods for the annotation of cell identity. These include:

- **Accuracy**: total proportion of correct predictions (correct annotations) made by the model;
- **Sensitivity**: (or rate of TRUE positives). Considering all the cells that were annotated under a cell type, what proportion was actually of that type?
- **Specificity**: (or rate of TRUE negatives). Considering all cells that were not annotated under a cell type, what proportion was not of that type?

- **F1-Score:** harmonic mean of the sensitivity and the specificity. The harmonic mean is used to balance the results in the presence of classes (cell types) with different abundances.

Accuracy and F1-Score are used to summarize the overall performance of a method across all cell types in a dataset, while sensitivity and specificity can be used to evaluate/compare the efficiency in the annotation of different cell types.

In 2019, Adelaal et al. evaluated the performances of 22 distinct methods for cell type annotation from scRNA-seq data by assessing their respective median F1-scores on a carefully constructed reference benchmark [140]. This benchmark was assembled to represent different use cases through the inclusion of datasets generated with different technologies and protocols, from different model organisms, and with varying levels of complexity. Each dataset included a different number (ranging from 1500 to 65000) of murine or human cells collected from pancreas, brain and blood samples and annotated at different levels of granularity. We refer the reader to the original study for a detailed description. Since its introduction the benchmark and evaluation criteria proposed by Abdelaal et al. have been widely adopted to compare and assess newly developed methods, but also for new benchmarking studies [141,142]. As a result these data are considered a golden standard in the field.

Table 6 collects the accuracies of 14 distinct tools discussed in this review for the classification of human pancreatic cells as reported by Abdelaal et al. and subsequent studies. While all methods demonstrated high accuracy, subtle differences can still be observed with some specific methods such as CaSTLe and SingleCellNet (C-ML), SingleR (SLL), sigGCN and scDeepHash (DL) or scMayoMap (MB) achieving an almost perfect median F1-score across all the datasets considered. However, considering the limited differences in performance, and the fact that tools based on different conceptual approaches yielded comparable results, it is hard to conclude that any of the five categories discussed in this review should have a definitive advantage.

2. Discussion and conclusion

This short review provides a concise and up to date summary of the most used computational approaches for the annotation of cellular identities in scRNA-seq. As clearly outlined in Tables 1–5 a considerable number of methods and analytical strategies have been developed in the last few years to tackle this problem. However, in this relatively young field, the quest for the development of the "one size fits all" definitive computational method to cell type annotation in scRNA-seq remains open. This is due to limitations both in the computational methods and in the data.

Indeed, any computational strategy discussed in this study presents important limitations that are yet to be solved. For example, DL needs extensive training data, is not easily human interpretable and requires dedicated hardware. C-ML and SSL are less computationally demanding compared to DL, but require well annotated/homogenised training sets -which are not currently available; additionally they need to be constantly updated and refined to recognise and classify previously unknown cell types. MB methods are more tractable and interpretable but require a very extensive biological knowledge which is probably beyond reach at the time being. HM tries to mitigate the limitation of each individual approach by integrating different algorithms into a single "consensus" prediction, however the efficient integration of prediction/classification from different methods is not always straightforward and represents a separate problem *per se*. Moreover the application of different methods fragments the decision problem and results in a loss of interpretability.

Table 6
F1-scores on pancreas benchmark data.

Tool	Category	Baron	Muraro	Xin	Segerstolpe
scMayoMap	MB	1.00	1.00	1.00	1.00
ACTINN	DL	0.98	0.97	0.95	1.00
Cell BLAST	DL	0.89	0.79	0.63	0.08
scBERT	DL	0.85	0.93	0.79	0.76
scDeepHash	DL	0.99	0.98	0.99	1.00
sigGCN	DL	0.97	1.00	0.99	1.00
scPred	C-ML	0.98	0.98	0.95	1.00
CaSTLe	C-ML	0.94	0.96	0.96	0.98
SingleCellNet	C-ML	0.96	0.97	1.00	0.99
CHEETAH	C-ML	0.94	0.96	0.96	0.97
scmapcell	SSL	0.98	0.97	0.73	1.00
scmapcluster	SSL	0.95	0.97	1.00	1.00
SingleR	SSL	0.97	0.95	0.99	0.97
ScID	SSL	0.59	0.95	0.80	0.85

Tool: tool name; **Category**: category, as defined in this review. Median F1-scores are reported in a distinct column for every dataset considered

From the data point of view, the relatively low resolution of single cell RNAseq, where only a handful of transcripts can be detected from each cell, is probably the most critical issue. The limit threshold of the technology for the detection of distinct cell types is not currently known and even worse, this limit might be different depending on the sequencing strategy and library preparation strategies. Another crucial limitation (see [20] for an in depth discussion) is the lack of a clear and universally accepted definition of "cell type". Indeed, notwithstanding the recent development of standardized ontologies and taxonomies for cell type representation, these resources do not completely match/reproduce the level of granularity of expert manual annotation and large inconsistencies in the annotation of the same cell-type or dataset can often be observed [143]. These ambiguities in classification do not provide an ideal framework for the development of computational methods, and especially for the development of ML-based applications where the "correct" labels are learned from the data.

In conclusion, large scale initiatives and coordinated efforts by the scientific community for the harmonisation/homogenization of analytical protocols and the interpretation of key findings - for example the Human Cell Atlas- represent a fundamental prerequisite for the development of the field, and will be decisive to aid in the development of highly accurate and comprehensive methods for the annotation of cellular identities and the future.

In the light of the above considerations and considering the open challenges in the field we believe we can offer the following advice to guide the reader in the selection of the most appropriate tool for cell identity annotation in their scRNA-seq data:

1. Make sure to follow best practices in the analyses of the data. Systematic differences/mistakes might systematically confound cell type annotation;
2. Check carefully which cell types can be natively annotated by a tool and favour tools that were developed/trained to annotate cell type of interest (adding custom configurations might be challenging for non expert and new users);
3. If/when available, execute exploratory analyses on publicly available data and replicate the results. First hand experience can greatly aid the selection of the "best" method for your application;
4. Clearly define the key objectives. Is explainability/interpretation of the model/classification important? If so, avoid overly complex models with several layers of abstraction. Would you like to replicate the annotation as in a specific study? Probably the best option is to select the same exact method;
5. Is customization required? At the time being probably MB based methods are more easily customized and adapted while training or modifying C-ML models might be more complex. DL methods cannot be modified/installed/configured by an end user. Dedicated hardware is required;
6. Manual annotation might still be required for the identification of novel cell types not represented in publicly available databases. Do not blindly trust automatic classification.

Statement

We the undersigned declare that this manuscript is original, has not been published before and is not currently being considered for publication elsewhere.

Funding and acknowledgements

This work was supported by Ministero dell'Università e della Ricerca, National Center for Gene Therapy and Drugs Based on RNA Technology [project number CN_00000041 to M.C.]; European Life-Science Infrastructure—Italy (ELIXIR-IT), ELIXIR x NextGenerationIT (ELIXIRxNextGenIT) project [grant code IR0000010 to M.C].

CRediT authorship contribution statement

Traversa Daniele: Writing – original draft, Investigation, Conceptualization. **Chiara Matteo:** Writing – review & editing, Supervision, Conceptualization.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Hooke, R. Micrographia: or some physiological descriptions of minute bodies made by magnifying glasses. With observations and inquiries thereupon. London: Printed by Jo. Martyn, and Ja. Allestry … and are to be sold at their shop. 1665.

[2] Xie B, Jiang Q, Mora A, Li X. Automatic cell type identification methods for single-cell RNA sequencing. Comput Struct Biotechnol J 2021;19:5874–87. https://doi.org/10.1016/j.csbj.2021.10.027.

[3] A. Regev *et al.*, The Human Cell Atlas, *eLife*, vol. 6, p. e27041, 2017, doi: 10.7554/eLife.27041.

[4] The Tabula Sapiens Consortium, The Tabula Sapiens: A multiple-organ, single-cell transcriptomic atlas of humans, *Science*, vol. 376, no. 6594, p. eabl4896, 2022, doi: 10.1126/science.abl4896.

[5] Baysoy A, Bai Z, Satija R, et al. The technological landscape and applications of single-cell multi-omics. Nat Rev Mol Cell Biol 2023;24:695–713. https://doi.org/10.1038/s41580-023-00615-w.

[6] Jovic D, Liang X, Zeng H, Lin L, Xu F, Luo Y. Single-cell RNA sequencing technologies and applications: A brief overview. Clin Transl Med 2022;12(3):e694. https://doi.org/10.1002/ctm2.694.

[7] Tang F, Barbacioru C, Wang Y, et al. mRNA-Seq whole-transcriptome analysis of a single cell. Nat Methods 2009;6(5):377–82. https://doi.org/10.1038/nmeth.1315.

[8] Ziegenhain C, et al. Comparative Analysis of Single-Cell RNA Sequencing Methods. Mol Cell Feb. 2017;65(4):631–643.e4. https://doi.org/10.1016/j.molcel.2017.01.023.

[9] Griffiths JA, Richard AC, Bach K, et al. Detection and removal of barcode swapping in single-cell RNA-seq data. Nat Commun 2018;9:2667. https://doi.org/10.1038/s41467-018-05083-x.

[10] Zheng GXY, et al. Massively parallel digital transcriptional profiling of single cells. Nat Commun 2017;8:1–12. https://doi.org/10.1038/ncomms14049.

[11] Smith T, Heger A, Sudbery I. UMI-tools: modeling sequencing errors in Unique Molecular Identifiers to improve quantification accuracy. Genome Res 2017;27(3):491–9.

[12] Satija R, Farrell JA, Gennert D, Schier AF, Regev A. Spatial reconstruction of single-cell gene expression data. Nat Biotechnol 2015;33:495–502. https://doi.org/10.1038/nbt.3192. ⟨https://doi.org/10.1038/nbt.3192⟩.

[13] Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. Genome Biol 2018;19(1):15. https://doi.org/10.1186/s13059-017-1382-0.

[14] Slovin S, et al. In: Picardi E, editor. Single-Cell RNA Sequencing Analysis: A Step-by-Step Overview, in RNA Bioinformatics. New York, NY: Springer US; 2021. p. 343–65. https://doi.org/10.1007/978-1-0716-1307-8_19.

[15] Heumos L, Schaar AC, Lance C, Litinetskaya A, Drost F, et al. Best practices for single-cell analysis across modalities. Nat Rev Genet 2023;24(8):550–72. https://doi.org/10.1038/s41576-023-00586-w.

[16] Germain PL, Sonrel A, Robinson MD. pipeComp, a general framework for the evaluation of computational pipelines, reveals performant single cell RNA-seq preprocessing tools. Genome Biol 2020;21(1):227. https://doi.org/10.1186/s13059-020-02136-7.

[17] Sun S, Zhu J, Ma Y, Zhou X. Accuracy, robustness and scalability of dimensionality reduction methods for single-cell RNA-seq analysis. Genome Biol 2019;20(1):269.

[18] Yu L, Cao Y, Yang JYH, Yang P. Benchmarking clustering algorithms on estimating the number of cell types from single-cell RNA-sequencing data. Genome Biol 2022;23(1):49. https://doi.org/10.1186/s13059-022-02622-0.

[19] Abdelaal T, et al. A comparison of automatic cell identification methods for single-cell RNA sequencing data. Genome Biol Sep. 2019;20(1):194. https://doi.org/10.1186/s13059-019-1795-z.

[20] Zeng H. What is a cell type and how to define it? Cell Jul. 2022;185(15):2739–55. https://doi.org/10.1016/j.cell.2022.06.031.

[21] Clarke ZA, et al. Tutorial: guidelines for annotating single-cell transcriptomic maps using automated and manual methods (Jun) Nat Protoc 2021;16(6):2749–64. https://doi.org/10.1038/s41596-021-00534-0.

[22] Pullin JM, McCarthy DJ. A comparison of marker gene selection methods for single-cell RNA sequencing data. Genome Biol 2024;25(1):56. https://doi.org/10.1186/s13059-024-03183-0.

[23] Greener JG, Kandathil SM, Moffat L, Jones DT. A guide to machine learning for biologists. Nat Rev *Mol Cell Biol* 2022;23(1):40–55. https://doi.org/10.1038/s41580-021-00407-0.

[24] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44. https://doi.org/10.1038/nature14539.

[25] Mnih V, et al. Human-level control through deep reinforcement learning. Nature Feb. 2015;518(7540):529–33. https://doi.org/10.1038/nature14236.

[26] Mohammadi S, Davila-Velderrain J, Kellis M. A multiresolution framework to characterize single-cell state landscapes. Nat Commun Oct. 2020;11(1):5399. https://doi.org/10.1038/s41467-020-18416-6.

[27] Franzén O, Björkegren JLM. alona: a web server for single-cell RNA-seq analysis. Bioinformatics Jun. 2020;36(12):3910–2. https://doi.org/10.1093/bioinformatics/btaa269.

[28] Mädler SC, et al. Besca, a single-cell transcriptomics analysis toolkit to accelerate translational research. NAR Genom Bioinforma Dec. 2021;3(4):lqab102. https://doi.org/10.1093/nargab/lqab102.

[29] I. Ferrari et al., CIA: a Cluster Independent Annotation method to investigate cell identities in scRNA-seq data, Aug. 26, 2024, bioRxiv. doi: 10.1101/2023.11.30.569382.

[30] Zhang AW, et al. Probabilistic cell-type assignment of single-cell RNA-seq for tumor microenvironment profiling. Nat Methods Oct. 2019;16(10):1007–15. https://doi.org/10.1038/s41592-019-0529-1.

[31] Mao S, Zhang Y, Seelig G, Kannan S. CellMeSH: probabilistic cell-type identification using indexed literature. Bioinformatics Mar. 2022;38(5):1393–402. https://doi.org/10.1093/bioinformatics/btab834.

[32] Pei G, Yan F, Simon LM, Dai Y, Jia P, Zhao Z. deCS: A Tool for Systematic Cell Type Annotations of Single-cell RNA Sequencing Data among Human Tissues. Genom, Proteom Bioinforma Apr. 2023;21(2):370–84. https://doi.org/10.1016/j.gpb.2022.04.001.

[33] Domanskyi S, Szedlak A, Hawkins NT, Wang J, Paternostro G, Piermarocchi C. Polled Digital Cell Sorter (p-DCS): Automatic identification of hematological cell types from single cell RNA-sequencing clusters. BMC Bioinforma Jul. 2019;20(1):369. https://doi.org/10.1186/s12859-019-2951-x.

[34] Kim H, Lee J, Kang K, Yoon S. MarkerCount: A stable, count-based cell type identifier for single-cell RNA-seq experiments. Comput Struct Biotechnol J Jan. 2022;20:3120–32. https://doi.org/10.1016/j.csbj.2022.06.010.

[35] Xu Y, Baumgart SJ, Stegmann CM, Hayat S. MACA: marker-based automatic cell-type annotation for single-cell expression data. Bioinformatics Mar. 2022;38(6):1756–60. https://doi.org/10.1093/bioinformatics/btab840.

[36] Nouri N, Gaglia G, Kurlovs AH, de Rinaldis E, Savova V. A marker gene-based method for identifying the cell-type of origin from single-cell RNA sequencing data. MethodsX Jan. 2023;10:102196. https://doi.org/10.1016/j.mex.2023.102196.

[37] Shao X, Liao J, Lu X, Xue R, Ai N, Fan X. scCATCH: Automatic Annotation on Cell Types of Clusters from Single-Cell RNA Sequencing Data. iScience Mar. 2020;23(3):100882. https://doi.org/10.1016/j.isci.2020.100882.

[38] Zhang Z, et al. SCINA: A Semi-Supervised Subtyping Algorithm of Single Cells and Bulk Samples. Art. no. 7 Genes Jul. 2019;10(7). https://doi.org/10.3390/genes10070531.

[39] Yang L, et al. Single-cell Mayo Map (scMayoMap): an easy-to-use tool for cell type annotation in single-cell RNA-sequencing data analysis. BMC Biol Oct. 2023;21(1):223. https://doi.org/10.1186/s12915-023-01728-6.

[40] Mikolajewicz N, Gacesa R, Aguilera-Uribe M, Brown KR, Moffat J, Han H. Multi-level cellular and functional annotation of single-cell transcriptomes using scPipeline. Commun Biol Oct. 2022;5(1):1–14. https://doi.org/10.1038/s42003-022-04093-2.

[41] Zhang Y, Zhang F, Wang Z, Wu S, Tian W. scMAGIC: accurately annotating single cells using two rounds of reference-based classification. Nucleic Acids Res May 2022;50(8):e43. https://doi.org/10.1093/nar/gkab1275.

[42] Li J, Sheng Q, Shyr Y, Liu Q. scMRMA: single cell multiresolution marker-based annotation. Nucleic Acids Res Jan. 2022;50(2):e7. https://doi.org/10.1093/nar/gkab931.

[43] Nassiri I, Fairfax B, Lee A, Wu Y, Buck D, Piazza P. scQCEA: a framework for annotation and quality control report of single-cell RNA-sequencing data. BMC Genom Jul. 2023;24(1):381. https://doi.org/10.1186/s12864-023-09447-6.

[44] Cao Y, Wang X, Peng G. SCSA: A Cell Type Annotation Tool for Single-Cell RNA-seq Data. Front Genet May 2020;11. https://doi.org/10.3389/fgene.2020.00490.

[45] Guo H, Li J. scSorter: assigning cells to known cell types according to marker genes. Genome Biol Feb. 2021;22(1):69. https://doi.org/10.1186/s13059-021-02281-7.

[46] Ianevski A, Giri AK, Aittokallio T. Fully-automated and ultra-fast cell-type identification using specific marker combinations from single-cell transcriptomic

data. Nat Commun Mar. 2022;13(1):1246. https://doi.org/10.1038/s41467-022-28803-w.

[47] Choi J-H, In Kim H, Woo HG. scTyper: a comprehensive pipeline for the cell typing analysis of single-cell RNA-seq data. BMC Bioinforma Aug. 2020;21(1): 342. https://doi.org/10.1186/s12859-020-03700-5.

[48] Pliner HA, Shendure J, Trapnell C. Supervised classification enables rapid annotation of cell atlases. Nat Methods Oct. 2019;16(10):983–6. https://doi.org/10.1038/s41592-019-0535-3.

[49] Hu C, et al. CellMarker 2.0: an updated database of manually curated cell markers in human/mouse and web tools based on scRNA-seq data. Nucleic Acids Res Jan. 2023;51(D1):D870–6. https://doi.org/10.1093/nar/gkac947.

[50] Franzén O, Gan L-M, Björkegren JLM. PanglaoDB: a web server for exploration of mouse and human single-cell RNA sequencing data. p. baz046 Database Jan. 2019;2019. https://doi.org/10.1093/database/baz046.

[51] Lieberman Y, Rokach L, Shay T. CaSTLe – Classification of single cells by transfer learning: Harnessing the power of publicly available single cell RNA sequencing experiments to annotate new experiments. PLOS ONE Oct. 2018;13(10): e0205499. https://doi.org/10.1371/journal.pone.0205499.

[52] de Kanter JK, Lijnzaad P, Candelli T, Margaritis T, Holstege FCP. CHETAH: a selective, hierarchical cell type identification method for single-cell RNA sequencing. Nucleic Acids Res Sep. 2019;47(16):e95. https://doi.org/10.1093/nar/gkz543.

[53] Bernstein MN, Ma Z, Gleicher M, Dewey CN. CellO: comprehensive and hierarchical cell type classification of human cells with the Cell Ontology. iScience Jan. 2021;24(1):101913. https://doi.org/10.1016/j.isci.2020.101913.

[54] K. Thorner, A.M. Zorn, and P. Chaturvedi, ELeFHAnt: A supervised machine learning approach for label harmonization and annotation of single cell RNA-seq data, Sep. 08, 2021, bioRxiv. doi: 10.1101/2021.09.07.459342.

[55] Kaymaz Y, et al. HieRFIT: a hierarchical cell type classification tool for projections from complex single-cell atlas datasets. Bioinformatics Dec. 2021;37 (23):4431–6. https://doi.org/10.1093/bioinformatics/btab499.

[56] K. Motwani, R. Bacher, and A.J. Molstad, Binned multinomial logistic regression for integrative cell type annotation, Nov. 23, 2021, arXiv: arXiv:2111.12149. doi: 10.48550/arXiv.2111.12149.

[57] Nguyen V, Griss J. scAnnotatR: framework to accurately classify cell types in single-cell RNA-sequencing data. BMC Bioinforma Jan. 2022;23(1):44. https://doi.org/10.1186/s12859-022-04574-5.

[58] Shen Y, Chu Q, Timko MP, Fan L. scDetect: a rank-based ensemble learning algorithm for cell type identification of single-cell RNA sequencing in cancer. Bioinformatics Nov. 2021;37(22):4115–22. https://doi.org/10.1093/bioinformatics/btab410.

[59] Li C, et al. SciBet as a portable and fast single cell type identifier. Nat Commun Apr. 2020;11(1):1818. https://doi.org/10.1038/s41467-020-15523-2.

[60] Alquicira-Hernandez J, Sathe A, Ji HP, Nguyen Q, Powell JE. scPred: accurate supervised method for cell-type classification from single-cell RNA-seq data. Genome Biol Dec. 2019;20(1):264. https://doi.org/10.1186/s13059-019-1862-5.

[61] Tan Y, Cahan P. SingleCellNet: A Computational Tool to Classify Single Cell RNA-Seq Data Across Platforms and Across Species. Cell Syst Aug. 2019;9(2):207–213. e2. https://doi.org/10.1016/j.cels.2019.06.004.

[62] F. Wagner and I. Yanai, Moana: A robust and scalable cell type classification framework for single-cell RNA-Seq data, Oct. 30, 2018, bioRxiv. doi: 10.1101/456129.

[63] Liu B, Wu F-X, Zou X. scASK: A Novel Ensemble Framework for Classifying Cell Types Based on Single-cell RNA-seq Data. IEEE J Biomed Health Inform Aug. 2021;25(8):3230–9. https://doi.org/10.1109/JBHI.2021.3050963.

[64] Dal Molin A, Di Camillo B. How to design a single-cell RNA-sequencing experiment: pitfalls, challenges and perspectives. Brief Bioinforma 2019;20(4): 1384–94. https://doi.org/10.1093/bib/bby007.

[65] Ekiz HA, Conley CJ, Stephens WZ, OConnell RM. CIPR: a web-based R/shiny app and R package to annotate cell clusters in single cell RNA sequencing experiments. BMC Bioinforma May 2020;21(1):191. https://doi.org/10.1186/s12859-020-3538-2.

[66] Kong W, et al. Capybara: A computational tool to measure cell identity and fate transitions. e11 Cell Stem Cell Apr. 2022;29(4):635–49. https://doi.org/10.1016/j.stem.2022.03.001.

[67] R. Fu *et al.*, clustifyr: an R package for automated single-cell RNA sequencing cluster classification, Jul. 16, 2020, F1000Research: 9:223. doi: 10.12688/f1000research.22969.2.

[68] Zanini F, et al. Northstar enables automatic classification of known and novel cell types from tumor samples. Sci Rep Sep. 2020;10(1):15251. https://doi.org/10.1038/s41598-020-71805-1.

[69] J. Gao, S. Guo, and Y. Zhang, ProjectSVR: Mapping single-cell RNA-seq data to reference atlases by supported vector regression, Aug. 02, 2023, bioRxiv. doi: 10.1101/2023.07.31.551202.

[70] Schmidt F, et al. RCA2: a scalable supervised clustering algorithm that reduces batch effects in scRNA-seq data. Nucleic Acids Res Sep. 2021;49(15):8505–19. https://doi.org/10.1093/nar/gkab632.

[71] Boufea K, Seth S, Batada NN. scID Uses Discriminant Analysis to Identify Transcriptionally Equivalent Cell Types across Single-Cell RNA-Seq Data with Batch Effect. iScience Mar. 2020;23(3):100914. https://doi.org/10.1016/j.isci.2020.100914.

[72] Kiselev VY, Yiu A, Hemberg M. scmap: projection of single-cell RNA-seq data across data sets. Nat Methods May 2018;15(5):359–62. https://doi.org/10.1038/nmeth.4644.

[73] Ekiz HA, Conley CJ, Stephens WZ, OConnell RM. CIPR: a web-based R/shiny app and R package to annotate cell clusters in single cell RNA sequencing experiments. BMC Bioinforma May 2020;21(1):191. https://doi.org/10.1186/s12859-020-3538-2.

[74] Kong W, et al. Capybara: A computational tool to measure cell identity and fate transitions. e11 Cell Stem Cell Apr. 2022;29(4):635–49. https://doi.org/10.1016/j.stem.2022.03.001.

[75] R. Fu et al., clustifyr: an R package for automated single-cell RNA sequencing cluster classification, Jul. 16, 2020, F1000Research: 9:223. doi: 10.12688/f1000research.22969.2.

[76] Zanini F, et al. Northstar enables automatic classification of known and novel cell types from tumor samples. Sci Rep Sep. 2020;10(1):15251. https://doi.org/10.1038/s41598-020-71805-1.

[77] J. Gao, S. Guo, and Y. Zhang, ProjectSVR: Mapping single-cell RNA-seq data to reference atlases by supported vector regression, Aug. 02, 2023, bioRxiv. doi: 10.1101/2023.07.31.551202.

[78] Schmidt F, et al. RCA2: a scalable supervised clustering algorithm that reduces batch effects in scRNA-seq data. Nucleic Acids Res Sep. 2021;49(15):8505–19. https://doi.org/10.1093/nar/gkab632.

[79] Zhou X, Chai H, Zeng Y, Zhao H, Yang Y. scAdapt: virtual adversarial domain adaptation network for single cell RNA-seq data classification across platforms and species. Brief Bioinforma Nov. 2021;22(6):bbab281. https://doi.org/10.1093/bib/bbab281.

[80] Boufea K, Seth S, Batada NN. scID Uses Discriminant Analysis to Identify Transcriptionally Equivalent Cell Types across Single-Cell RNA-Seq Data with Batch Effect. iScience Mar. 2020;23(3):100914. https://doi.org/10.1016/j.isci.2020.100914.

[81] Duan B, et al. Learning for single-cell assignment. Sci Adv Oct. 2020;6(44): eabd0855. https://doi.org/10.1126/sciadv.abd0855.

[82] Kiselev VY, Yiu A, Hemberg M. scmap: projection of single-cell RNA-seq data across data sets. Nat Methods May 2018;15(5):359–62. https://doi.org/10.1038/nmeth.4644.

[83] Hou R, Denisenko E, Forrest ARR. scMatch: a single-cell gene expression profile annotation tool using reference datasets. Bioinformatics Nov. 2019;35(22): 4688–95. https://doi.org/10.1093/bioinformatics/btz292.

[84] Aran D, et al. Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. Nat Immunol Feb. 2019;20(2):163–72. https://doi.org/10.1038/s41590-018-0276-y.

[85] Deng Y, Choi J, Lê Cao K-A. Sincast: a computational framework to predict cell identities in single-cell transcriptomes using bulk atlases as references. Brief Bioinforma May 2022;23(3):bbac088. https://doi.org/10.1093/bib/bbac088.

[86] Y. Wang et al., Automated single-cell omics end-to-end framework with data-driven batch inference, Jun. 20, 2024, bioRxiv. doi: 10.1101/2023.11.01.564815.

[87] Kang JB, et al. Efficient and precise single-cell reference atlas mapping with Symphony. Nat Commun Oct. 2021;12(1):5890. https://doi.org/10.1038/s41467-021-25957-x.

[88] Ma F, Pellegrini M. ACTINN: automated identification of cell types in single cell RNA sequencing. Bioinformatics Jan. 2020;36(2):533–8. https://doi.org/10.1093/bioinformatics/btz592.

[89] Li Z, Wang Y, Ganan-Gomez I, Colla S, Do K-A. A machine learning-based method for automatically identifying novel cells in annotating single-cell RNA-seq data. Bioinformatics Nov. 2022;38(21):4885–92. https://doi.org/10.1093/bioinformatics/btac617.

[90] Yin Q, Chen L. CellTICS: an explainable neural network for cell-type identification and interpretation based on single-cell RNA-seq data. Brief Bioinforma Jan. 2024; 25(1):bbad449. https://doi.org/10.1093/bib/bbad449.

[91] Cao Z-J, Wei L, Lu S, Yang D-C, Gao G. Searching large-scale scRNA-seq databases via unbiased cell embedding with Cell BLAST. Nat Commun Jul. 2020;11(1): 3458. https://doi.org/10.1038/s41467-020-17281-7.

[92] Xu J, Zhang A, Liu F, Chen L, Zhang X. CIForm as a Transformer-based model for cell-type annotation of large-scale single-cell RNA-seq data. Brief Bioinforma Jul. 2023;24(4):bbad195. https://doi.org/10.1093/bib/bbad195.

[93] Goyal M, Serrano G, Argemi J, Shomorony I, Hernaez M, Ochoa I. JIND: joint integration and discrimination for automated single-cell annotation. Bioinformatics Apr. 2022;38(9):2488–95. https://doi.org/10.1093/bioinformatics/btac140.

[94] Xiong Y-X, Wang M-G, Chen L, Zhang X-F. Cell-type annotation with accurate unseen cell-type identification using multiple references. PLOS Comput Biol Jun. 2023;19(6):e1011261. https://doi.org/10.1371/journal.pcbi.1011261.

[95] Duan B, et al. Integrating multiple references for single-cell assignment. Nucleic Acids Res Aug. 2021;49(14):e80. https://doi.org/10.1093/nar/gkab380.

[96] A.A. Heydari, O.A. Davalos, K.K. Hoyer, and S.S. Sindi, N-ACT: An Interpretable Deep Learning Model for Automatic Cell Type and Salient Gene Identification, May 08, 2022, arXiv: arXiv:2206.04047. doi:10.48550/arXiv.2206.04047.

[97] Li Z, Feng H. A neural network-based method for exhaustive cell label assignment using single cell RNA-seq data. Sci Rep Jan. 2022;12(1):910. https://doi.org/10.1038/s41598-021-04473-4.

[98] S. Zhao, J. Zhang, and Z. Nie, Large-Scale Cell Representation Learning via Divide-and-Conquer Contrastive Learning, Jun. 07, 2023, arXiv: arXiv:2306.04371. doi: 10.48550/arXiv.2306.04371.

[99] Wang S, et al. Leveraging the Cell Ontology to classify unseen cell types. Nat Commun Sep. 2021;12(1):5556. https://doi.org/10.1038/s41467-021-25725-x.

[100] Storrs EP, et al. Pollock: fishing for cell states. Bioinforma Adv Jan. 2022;2(1): vbac028. https://doi.org/10.1093/bioadv/vbac028.

[101] O.A. Davalos, A.A. Heydari, E.J. Fertig, S.S. Sindi, and K.K. Hoyer, Boosting Single-Cell RNA Sequencing Analysis with Simple Neural Attention, Jun. 01, 2023, bioRxiv. doi: 10.1101/2023.05.29.542760.

[102] Yang F, et al. scBERT as a large-scale pretrained deep language model for cell type annotation of single-cell RNA-seq data. Nat Mach Intell Oct. 2022;4(10):852–66. https://doi.org/10.1038/s42256-022-00534-z.

[103] Wang L, et al. An interpretable deep-learning architecture of capsule networks for identifying cell-type gene expression programs from single-cell RNA-sequencing data. Nat Mach Intell Nov. 2020;2:1–11. https://doi.org/10.1038/s42256-020-00244-4.

[104] S. Ma, Y. Zhang, B. Wang, Z. Hu, J. Zhang, and B. Wang, scDeepHash: An automatic cell type annotation and cell retrieval method for large-scale scRNA-seq datasets using neural network-based hashing, Nov. 10, 2021, bioRxiv. doi: 10.1101/2021.11.08.467820.

[105] Jia S, Lysenko A, Boroevich KA, Sharma A, Tsunoda T. scDeepInsight: a supervised cell-type identification method for scRNA-seq data with deep learning. Brief Bioinforma Sep. 2023;24(5):bbad266. https://doi.org/10.1093/bib/bbad266.

[106] Shao X, et al. scDeepSort: a pre-trained cell-type annotation method for single-cell transcriptomics using deep learning with a weighted graph neural network. Nucleic Acids Res Dec. 2021;49(21):e122. https://doi.org/10.1093/nar/gkab775.

[107] Tran D, Nguyen H, Tran B, La Vecchia C, Luu HN, Nguyen T. Fast and precise single-cell data analysis using a hierarchical autoencoder. Nat Commun Feb. 2021;12(1):1029. https://doi.org/10.1038/s41467-021-21312-2.

[108] Yin Q, et al. scGraph: a graph neural network-based approach to automatically identify cell types. Bioinformatics May 2022;38(11):2996–3003. https://doi.org/10.1093/bioinformatics/btac199.

[109] Yuan M, Chen L, Deng M. scMRA: a robust deep learning method to annotate scRNA-seq data with multiple reference datasets. Bioinformatics Jan. 2022;38(3): 738–45. https://doi.org/10.1093/bioinformatics/btab700.

[110] Zhang R, Luo Y, Ma J, Zhang M, Wang S. scPretrain: multi-task self-supervised learning for cell-type classification. Bioinformatics Mar. 2022;38(6):1607–14. https://doi.org/10.1093/bioinformatics/btac007.

[111] Y. Liu et al., scRCA: a Siamese network-based pipeline for the annotation of cell types using imperfect single-cell RNA-seq reference data, Apr. 11, 2024, bioRxiv. doi: 10.1101/2024.04.08.588510.

[112] Lewinsohn DP, Vigh-Conrad KA, Conrad DF, Scott CB. Consensus label propagation with graph convolutional networks for single-cell RNA sequencing cell type annotation. Bioinformatics Jun. 2023;39(6):btad360. https://doi.org/10.1093/bioinformatics/btad360.

[113] Ren P, et al. Single-cell assignment using multiple-adversarial domain adaptation network with large-scale references. Cell Rep Methods Sep. 2023;3(9):100577. https://doi.org/10.1016/j.crmeth.2023.100577.

[114] Wang T, Bai J, Nabavi S. Single-cell classification using graph convolutional networks. BMC Bioinforma Jul. 2021;22(1):364. https://doi.org/10.1186/s12859-021-04278-2.

[115] Gonzalez-Ferrer J, et al. SIMS: A deep-learning label transfer tool for single-cell RNA sequencing analysis. Cell Genom Jun. 2024;4(6):100581. https://doi.org/10.1016/j.xgen.2024.100581.

[116] Xie P, et al. SuperCT: a supervised-learning framework for enhanced characterization of single-cell transcriptomic profiles. Nucleic Acids Res May 2019;47(8):e48. https://doi.org/10.1093/nar/gkz116.

[117] Chen J, Xu H, Tao W, Chen Z, Zhao Y, Han J-DJ. Transformer for one stop interpretable cell type annotation. Nat Commun Jan. 2023;14(1):223. https://doi.org/10.1038/s41467-023-35923-4.

[118] Cui H, et al. scGPT: toward building a foundation model for single-cell multi-omics using generative AI. Nat Methods Aug. 2024;21(8):1470–80. https://doi.org/10.1038/s41592-024-02201-0.

[119] Brbić M, et al. MARS: discovering novel cell types across heterogeneous single-cell experiments. Nat Methods Dec. 2020;17(12):1200–6. https://doi.org/10.1038/s41592-020-00979-3.

[120] Ma Q, Xu D. Deep learning shapes single-cell data analysis. Nat Rev Mol Cell Biol May 2022;23(5):303–4. https://doi.org/10.1038/s41580-022-00466-x.

[121] Szałata A, Hrovatin K, Becker S, Tejada-Lapuerta A, Cui H, Wang B, Theis FJ. Transformers in single-cell omics: a review and new perspectives. Nat Methods 2024;21:1430–43. https://doi.org/10.1038/s41592-024-02353-z.

[122] Passemiers A, Folco P, Raimondi D, et al. A quantitative benchmark of neural network feature selection methods for detecting nonlinear signals. Sci Rep 2024; 14:31180. https://doi.org/10.1038/s41598-024-82583-5.

[123] Chen V, Yang M, Cui W, Kim JS, Talwalkar A, Ma J. Applying interpretable machine learning in computational biology-pitfalls, recommendations and opportunities for new developments. Nat Methods 2024;21(8):1454–61. https://doi.org/10.1038/s41592-024-02359-7.

[124] Zeng Y, Wei Z, Pan Z, Lu Y, Yang Y. A robust and scalable graph neural network for accurate single-cell classification. Brief Bioinforma Mar. 2022;23(2):bbab570. https://doi.org/10.1093/bib/bbab570.

[125] Johnson TS, et al. LAmbDA: label ambiguous domain adaptation dataset integration reduces batch effects and improves subtype detection. Bioinformatics Nov. 2019;35(22):4696–706. https://doi.org/10.1093/bioinformatics/btz295.

[126] Yin Q, Wang Y, Guan J, Ji G. scIAE: an integrative autoencoder-based ensemble classification framework for single-cell RNA-seq data. Brief Bioinforma Jan. 2022; 23(1):bbab508. https://doi.org/10.1093/bib/bbab508.

[127] Kimmel JC, Kelley DR. Semisupervised adversarial neural networks for single-cell classification. Genome Res Jan. 2021;31(10):1781–93. https://doi.org/10.1101/gr.268581.120.

[128] Liu Y, et al. TripletCell: a deep metric learning framework for accurate annotation of cell types at the single-cell level. Brief Bioinforma May 2023;24(3):bbad132. https://doi.org/10.1093/bib/bbad132.

[129] Sun Q, Peng Y, Liu J. A reference-free approach for cell type classification with scRNA-seq. iScience Aug. 2021;24(8):102855. https://doi.org/10.1016/j.isci.2021.102855.

[130] Li D, Ding J, Bar-joseph Z. UNIFAN: A Tool for Unsupervised Single-Cell Clustering and Annotation. J Comput Biol Nov. 2022;29(11):1229–32. https://doi.org/10.1089/cmb.2022.0251.

[131] Chen L, Zhai Y, He Q, Wang W, Deng M. Integrating Deep Supervised, Self-Supervised and Unsupervised Learning for Single-Cell RNA-seq Clustering and Annotation. Art. no. 7 Genes Jul. 2020;11(7). https://doi.org/10.3390/genes11070792.

[132] E. Busarello *et al.*, Interpreting single-cell messages in normal and aberrant hematopoiesis with the Cell Marker Accordion, Mar. 12, 2024, bioRxiv. doi: 10.1101/2024.03.08.584053.

[133] Huang X, Liu R, Yang S, Chen X, Li H. scAnnoX: an R package integrating multiple public tools for single-cell annotation. PeerJ Mar. 2024;12:e17184. https://doi.org/10.7717/peerj.17184.

[134] Chen Y, Lakshmikanth T, Mikes J, Brodin P. bioRxiv. Single-Cell Classif Using Learn Cell phenotypes Jul. 24, 2020. https://doi.org/10.1101/2020.07.22.216002.

[135] Yang X, Gao S, Wang T, Yang B, Dang N, Ye K. gCAnno: a graph-based single cell type annotation method. BMC Genom Nov. 2020;21(1):823. https://doi.org/10.1186/s12864-020-07223-4.

[136] C. Ergen et al., Consensus prediction of cell type labels with popV, Aug. 21, 2023, bioRxiv. doi: 10.1101/2023.08.18.553912.

[137] Lin Y, et al. scClassify: sample size estimation and multiscale classification of cells using single and multiple reference. Mol Syst Biol Jun. 2020;16(6):e9389. https://doi.org/10.15252/msb.20199389.

[138] Michielsen L, Reinders MJT, Mahfouz A. Hierarchical progressive learning of cell identities in single-cell data. Nat Commun May 2021;12(1):2799. https://doi.org/10.1038/s41467-021-23196-8.

[139] Hao Y, Hao S, Andersen-Nissen E, et al. Integrated analysis of multimodal single-cell data. e29 Cell 2021;184(13):3573–87. https://doi.org/10.1016/j.cell.2021.04.048.

[140] Abdelaal T, et al. A comparison of automatic cell identification methods for single-cell RNA sequencing data. Art. no. 1 Genome Biol Sep. 2019;20(1). https://doi.org/10.1186/s13059-019-1795-z.

[141] Sun X, Lin X, Li Z, Wu H. A comprehensive comparison of supervised and unsupervised methods for cell type identification in single-cell RNA-seq. Brief Bioinforma 2022;23(2):bbab567. https://doi.org/10.1093/bib/bbab567.

[142] Fu Q, Dong C, Liu Y, et al. A comparison of scRNA-seq annotation methods based on experimentally labeled immune cell subtype dataset. Brief Bioinform 2024;25 (5):bbae392. https://doi.org/10.1093/bib/bbae392.

[143] Nisoli, E., & Cinti, S. (2024). What defines a cell type? Perspectives from adipocyte biology. International journal of obesity (2005), 10.1038/s41366-024-01696-z. Advance online publication. https://doi.org/10.1038/s41366-024-01696-z.